

# APLIKASI METODE TOP DOWN PARSING PADA GAME PEMBELAJARAN CISCO ROUTER

**Ahmad Syamsudin**

Universitas Nisantara PGRI Kediri  
Jl. K.H Ahmad Dahlan No. 76 Mojoroto Kediri, Jawa Timur  
email : udinulis@yahoo.com

## Abstrak

Kendala pelajar dalam mempelajari jaringan khususnya router cisco adalah mahalnya alat yang digunakan. Game edukasi dengan metode top-down parsing dapat menjadi solusi dalam mempelajari jaringan. Kelebihan game ini adalah tersedianya simulasi terbimbing dalam game. Tujuan penelitian ini adalah untuk membuat game edukasi berbasis text adventure agar pembelajaran jaringan mudah dimengerti dan menyenangkan. Penelitian ini memanfaatkan game engine GTGE sebagai framework system dengan metode top-down parsing sebagai penganalisa sintaksnya. Hasil penelitian ini menunjukkan bahwa pemanfaatan top-down parsing dapat digunakan untuk menganalisa teks inputan game. Sedangkan GTGE kurang tepat digunakan, karena tidak mendukung JTextArea yang mutlak digunakan untuk game yang berbasis text.

## Kata kunci :

*educational games, text adventure games, GTGE, cisco, top-down parsing*

## 1. Pendahuluan

Pembelajaran adalah upaya menciptakan kondisi dengan sengaja agar tujuan pembelajaran dapat dipermudah pencapaiannya [3]. Dalam kegiatan pembelajaran perlu dipilih strategi dan media yang tepat sehingga pelajar dapat mengoptimalkan ilmu yang dipelajarinya. Dalam konsep teknologi pendidikan, pemanfaatan media pembelajaran memiliki peranan penting dalam kegiatan belajar mengajar. Sudirdjo, Dalam Mozaik Pendidikan menjelaskan beberapa fungsi media komunikasi dalam pembelajaran yang diantaranya adalah memberikan pengalaman simulasi. Simulasi dapat menciptakan lingkungan buatan untuk merangsang siswa bereaksi terhadap materi yang diberikan secara audio visual. Lebih lanjut beliau membagi beberapa simulasi yang digunakan sebagai media pembelajaran diantaranya *visual educational, audiovisual education, educational communication, komik pendidikan dan edutainment game*.

Titik fokus dalam penelitian ini adalah pembelajaran setting cisco router menggunakan syntax command line IOS. Syntax tersebut akan dimasukkan kedalam game dua dimensi yang dimainkan secara single player. Pada awal permainan, pemain akan ditugaskan untuk memberikan IP address dan subnetmask pada

komputernya dan menganalisa jaringan agar dapat terkoneksi dengan server. Untuk mendapatkan IP address dan subnetmask, pemain harus memecahkan persoalan masalah subnetting yang berupa teka-teki angka biner dan diacak menurut kata kunci tertentu. Pemain akan mendapat reward berupa uang dalam bentuk alamat bank berikut kode aksesnya, jika berhasil menyelesaikan misi. Uang tersebut dapat digunakan untuk men-upgrade hardware baru demi kelancaran misi selanjutnya. Secara keseluruhan, pemain harus menggunakan command line untuk menyelesaikan setiap misi. Hal ini agar penguasaan pemain terhadap IOS cisco router meningkat seiring dengan selesainya misi tersebut, sehingga pemain akan mudah beradaptasi jika dihadapkan pada setting router cisco yang sesungguhnya.

### 1.1 Tujuan dan Manfaat Penelitian

Tujuan yang ingin dicapai dalam penelitian ini yaitu merancang dan membangun game edukasi untuk pembelajaran setting jaringan cisco router menggunakan metode topdown parsing. Manfaat yang diperoleh dari penggunaan game ini adalah untuk menambah pengetahuan dan pemahaman pemain dalam mensetting jaringan khususnya cisco router.

### 1.2 Batasan Penelitian

Peneliti menganggap user atau pemain game ini sudah memiliki pengetahuan dasar jaringan. Materi router Cisco yang ada dalam game ini adalah materi subnetting, routing, pengetahuan remote telnet, remote SSH, NAT dan Access list standard [2]. Script yang diinputkan oleh pemain tidak semuanya berupa listing Cisco, tetapi juga terdapat script tambahan.

## 2. Tinjauan Pustaka

### 2.1 Genre

Meigs, dalam bukunya *Ultimate Game Design* mengelompokkan genre game yang berkembang saat ini menjadi beberapa kelompok yaitu: Sport Games, Fighting Games, Puzzle Games, Role Playing Games, First Or Third person action Games Simulation Oriented Games dan Text Based Adventure Games (Jenis ini adalah game yang berbasis pada input keyboard pemain [1]. Contoh dari game ini adalah Zorg I, II, III, The Dungeon dan Hacker Evolution)

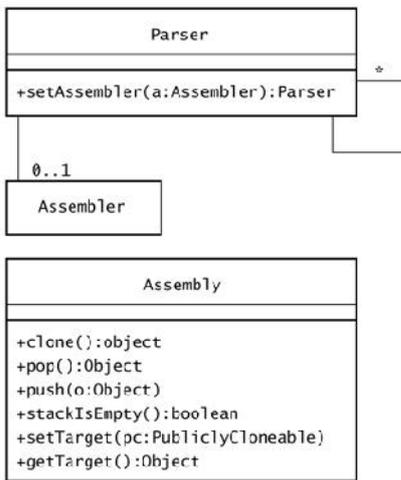
**2.2 Design Overview**

Dalam membangun parser, langkah pertama yang harus dipikirkan adalah beberapa contoh string yang ingin dikenalkan oleh parser tersebut. Untuk memparse kumpulan string tersebut, maka dibuatlah sebuah language (bahasa) baru. Language tersebut merupakan kumpulan string yang terdiri dari rules dan grammer [4].

Parser yang sudah dibuat akan mengenali string dan mengikuti rules dari grammer yang ada. Aspek utama yang lain dari pembangunan sebuah parser yaitu desain assembler. Assembler berguna untuk membuat objek baru ketika parser mengenali inputan string. Assembler akan memilih dan memilah sesuai aturan karakter mana yang akan dikategorikan menjadi number, huruf, karakter atau akan dibuang [5].

**2.2.1 Desain Assembler**

Sebuah assembly menyediakan stack dan target dari objek untuk parser. Objek target akan membentuk parser yang dapat mengenali inputan berupa teks. Berikut bentuk dari class assembler dijelaskan pada gambar 1 [4]



Gambar 1. Class Parser, Assembler dan Assembly

**2.2.2 Grammer**

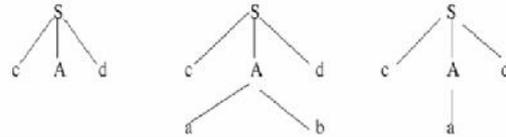
Grammer adalah kumpulan dari beberapa definisi parser yang saling berhubungan dan mengikuti standard aturan penulisan token yang ada. Tujuan akhir dari desain grammer adalah untuk membentuk dan mengetahui fitur-fitur apa saja yang harus ada dalam sebuah parser hingga parser tersebut dapat mengenali inputan teks String

Grammer parsing memiliki dua metode yang keduanya biasa digunakan sebagai syntax analyzer suatu bahasa pemrograman yaitu metode Bottom-Up Parsing dan metode Top-down Parsing.

Top-down parsing digunakan sebagai upaya menemukan penurunan terkecil string masukan. Parsing ini juga dapat dipandang sebagai upaya membangun pohon parse string masukan yang dimulai dari puncak dan menciptakan simpul-simpul pohon parse secara preorder seperti dijelaskan di gambar 2 berikut ini [7].

Contoh :

S => cAd  
 A => ab | a



Gambar 2: Bagan Pohon Syntax

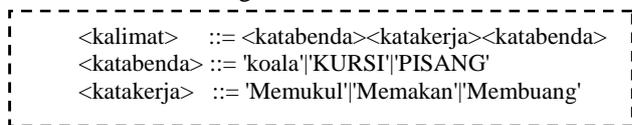
Metode topdown parsing ini memiliki beberapa algoritma yang terdiri dari:

- a. Brute Force Method  
 Brute Force Method berusaha menemukan turunan terkecil dari string masukan. Jika string yang dihasilkan tidak cocok dengan string yang diperiksa maka dilakukan backtracking.

- b. Recursive Descent Parsing  
 Penggunaan metode ini dimulai dari symbol awal yang dicoba memproduksi kalimat menggunakan produksi yang tepat [4:285]. Syarat grammar yang dapat diparse dengan recursive Descent Parsing adalah Context-free grammar, tidak mempunyai left recursion, menerapkan dulu alternative aturan produksi yang terpanjang (bila terdapat lebih dari satu alternative aturan produksi)

- c. Predictive Top-Down Parsing  
 Parsing ini juga disebut top-down parsing without backup dan deterministic top down parsing, menggunakan kelas grammar LL(k) sehingga disebut LL(k) parser. Kelas grammar LL(k) dapat diparse secara deterministik dengan melihat symbol masukan berikutnya yang akan diparse. Algoritma parsing untuk kelas grammar ini efisien untuk digunakan. Kinerja algoritma ini fungsi linear panjang string masukan. Teknik deteksi dan pemulihan kesalahan dapat secara mudah ditambahkan ke algoritma parsing ini.

Top down parsing juga dapat dijelaskan dengan contoh lain sebagai berikut Contoh :



Top down parser berusaha mengekspansi aturan produksi, dan mencocokkannya dengan input. Parser jenis ini akan mencoba mengekspansi <kalimat> menjadi <aksi> atau <pernyataan>, pertama jenis <aksi> akan dicoba (nantu jika ternyata bagian ini gagal, bagian <pernyataan> akan dicoba). Dari <aksi> bisa diekspansi menjadi <katabenda><katakerja><katasifat>. Kata benda diekspansi menjadi KOALA, KURSI, atau PISANG. Ternyata input KOALA cocok dengan salah satu terminal tersebut, sehingga bisa disimpulkan untuk saat ini, bahwa KOALA adalah <katabenda>

Ketika ingin mencocokkan ekspresi, parser akan mencoba mengekspansi <ekspresi> menjadi <ekspresi>

dan <ekspresi> menjadi <ekspresi>, dan demikian seterusnya. Masalah ini bisa diselesaikan dengan mengubah bentuk di atas menjadi bentuk non rekursif [7]

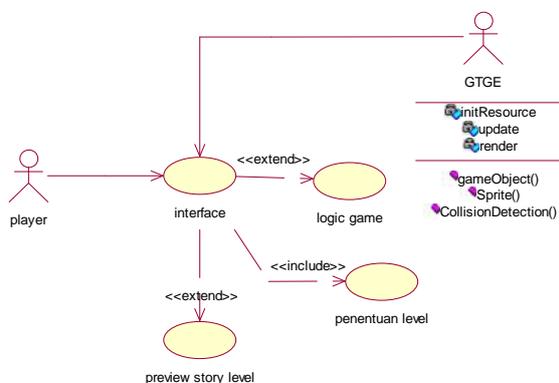
### 3. Metode Penelitian

#### 3.1 Desain Level dan Skenario

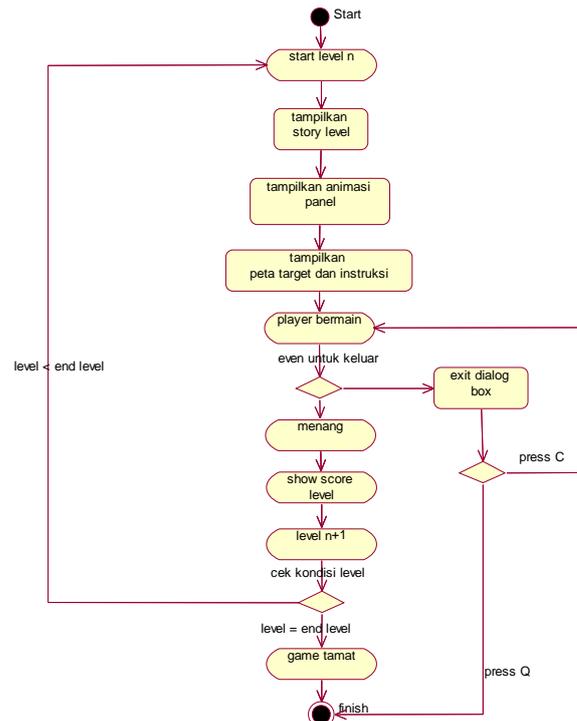
Level yang disediakan pada ini terdiri dari tiga level. Pada level pertama adalah level tutorial, pemain akan dikenalkan environment (lingkungan) dari game dan cara menggunakan sintaks dan tombol-tombol yang mendukung aktifitas dalam game oleh agen decken. Selain itu juga ada materi Cisco yang disertakan dalam level ini yaitu tentang subnetting, routing dan koneksi dengan telnet.

Level kedua, pemain akan diberi tugas untuk mensetting telnet, SSH dan routing. Dalam level ini pemain harus mampu memecahkan teka-teki atau persoalan terkait dengan jaringan agar dapat menyelesaikan misi. Level terakhir, pemain akan mencoba mengimplementasikan setting NAT, dan access list menggunakan koneksi SSH. Pemain akan mendapatkan materi tentang NAT dan access list setelah berhasil membobol komputer lawan.

Dalam game ini hacker juga akan mendapatkan uang sebagai bayaran atas jasanya mengungkap koruptor. Uang tersebut dapat digunakan untuk meng-upgrade spesifikasi hardware yang dimiliki seperti modem, memory, hardisk dan CPU setelah mendapat uang dari pemerintah yang diwakili oleh agen bernama decken. Dengan mengkombinasikan antara keunikan pembelajaran setting router Cisco dan ketegangan yang disuguhkan dalam game komersial “-hacker Evolution”, game ini diharapkan tetap menjadi game yang menarik dan menyenangkan untuk dimainkan. Alur cerita dan perpindahan yang terjadi antar tahap dalam game tersebut akan digambarkan dalam diagram use case gambar 3. dan juga didetailkan dalam dalam bentuk activity diagram yang terbagi menjadi tiga bagian. Masing-masing bagian mempunyai alur cerita yang saling berhubungan yang dijelaskan pada gambar 4



Gambar 3: Digram use case pada perpindahan level game



Gambar 4: Activity diagram perpindahan level game

Dari activity diagram pada gambar 4 diketahui bahwa game akan terus berjalan dari level pertama ke level selanjutnya. Program akan mengecek nilai level setiap kali pemain menyelesaikan misi. Jika nilai level sudah sama dengan nilai pada level terakhir, maka program akan berhenti. Sehingga untuk keluar dari permainan meski game belum selesai, pemain dapat menekan tombol escape (ESC) sewaktu-waktu.

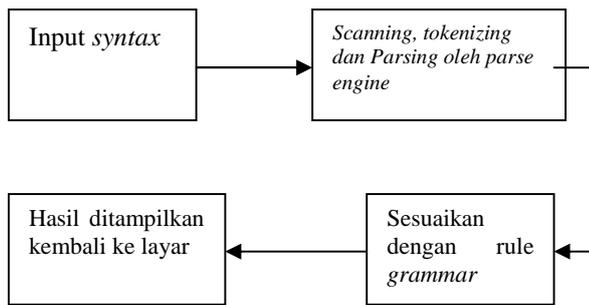
#### 3.2 Pemaparan Input Output

Adapun input, proses dan output yang terjadi dalam frame utama dapat dijelaskan secara singkat sebagai berikut:

##### a. Input

Untuk memainkan game edukasi ini user harus menggunakan inputan berupa teks atau perintah command line yang disesuaikan dengan standard IOS CISCO router versi 10.3 ke atas. Selain syntax IOS terdapat juga syntax command line fiktif yang dibuat untuk menyesuaikan scenario game.

Pemain juga akan memasukkan inputan berupa teks untuk mengisi profile pemain ketika game sudah berakhir. Berikut ini dijelaskan alur dari teks mulai dari input hingga ditampilkan hasilnya kembali dan dipaparkan pada gambar 5 berikut:



Gambar 5: Alur Input Data

b. Proses

Proses yang terjadi pada game ini dapat diklasifikasikan menjadi dua golongan.

- Proses Pengolahan input command line  
 Pada proses ini command line yang dimasukkan oleh pemain akan dipotong menjadi token, kemudian dianalisa dan dicocokkan dengan aturan penulisan syntax IOS Cisco router yang benar. Selain aturan penulisan sintaks IOS Cisco router, game ini juga menggunakan aturan penulisan sintaks bantuan yang dibuat untuk mempermudah proses -hacking. Dalam setiap sintaks yang dimasukkan oleh pemain setelah dianalisa dan dicocokkan, jika sesuai dengan aturan dan privilege usernya, maka pemain akan dapat memilih perintah berikutnya sesuai dengan sub perintah yang digunakan
  - Proses Skoring (Penghitungan Nilai)  
 Dalam proses ini skor pemain akan dihitung berdasarkan pada jumlah poin alamat yang berhasil dibobol dalam satu level, tingkat kesulitan tiap level dan waktu yang dibutuhkan oleh pemain untuk menyelesaikan level. Selain itu ada juga variable bonus menambah skor bagi pemain yang mampu menemukan misi tambahan dan menyelesaikannya.  
 Dari semua variable tersebut akan diberikan sebuah nilai, kemudian dihitung nilai rata-ratanya sehingga didapatkan total skor untuk pemain. Berikut rencana perhitungan skor bagi pemain  
 Variabel yang dihitung:  
 Jumlah alamat = JA (bernilai 20 poin untuk setiap alamat)  
 Tingkat kesulitan level = TK (bernilai 10 dan kelipatannya)  
 Waktu =W (nilai poin sudah dijelaskan di atas)  
 Misi tambahan = MT (Bonus, bernilai 30 poin)
- Sehingga penghitungan total skor adalah:
- Total skor =**  
**(JA x 20) + (TK x 10) + W + (MT x 30)**  
 Total skor di atas akan muncul secara otomatis setelah pemain menyelesaikan satu level. Syarat untuk maju ke level berikutnya adalah pemain

harus dapat menyelesaikan semua poin misi dalam satu level.

- c. Output  
 Output dari game ini dibedakan menjadi dua yaitu:
  - Output dari command line  
 Output ini berupa sintaks yang dimasukkan pemain, report hasil masukan sintaks, report jumlah alamat yang berhasil di -hack dan misi yang sudah dilaksanakan.
  - Output hasil Scoring  
 Output ini berupa skor yang didapat pemain setelah menyelesaikan misi, jumlah uang yang berhasil dikumpulkan pemain dan waktu yang dibutuhkan untuk menyelesaikan misi.

3.3 Desain Top-Down Grammar

Pendekatan yang dilakukan dalam mendesain top-down grammar yaitu adalah dengan mendesain masalah kedalam komponen[7]. Masalah yang ada dalam hal ini adalah sintaks game, dapat dipecah dan pisahkan menjadi beberapa komponen berupa token. Langkah yang digunakan dalam mendesain parser menggunakan metode ini yang pertama adalah dengan mendefinisikan parser yang diinginkan menjadi komponen dari subparser. Langkah kedua adalah mengulangi langkah pertama sampai setiap subparser terdefiniskan. Proses pendefinisian rule grammar terhadap sintaks dijelaskan dengan sebagai berikut.

Sebagai contoh game dalam keadaan belum terkoneksi (*not connected state*) harus dapat mengenali tiga perintah utama seperti gambar 3.21.

```

    Crack [target] = crack core.counter.com
    Scan [target] = scan core.counter.com
    Upgrade [target] = upgrade memory1
    
```

Gambar 6: Contoh Sintaks Game

Untuk melakukan proses parsing pada ketiga command diatas pertama ditentukan dahulu inialisasi grammar yang ingin dibuat yaitu *notConnected grammar*. Setelah inialisasi selesai, selanjutnya ditentukan parser dari *grammar notConnected* tersebut. Maka ditentukan rule pertama dari *grammar* yaitu:

```

    notConnected = crackCommand | scanCommand | upgradeCommand
    
```

Kemudian diperluas kearah kanan definisi dari rule diatas. Sebagai contoh untuk parser <crack>. Terdapat dua potong komponen dalam kode sintaks di atas yaitu [crack] dan [core.counter.com]. crack menjadi valid command word pertama dari sintaks, karena crack sudah terdaftar dalam rule grammar. Kemudian core.counter.com menjadi second word dari valid command word dan didefinisikan sebagai target.  
 notConnected = crackCommand | scanCommand | upgradeCommand  
 crackCommand = "crack" "target".

Dari perintah diatas parser <crack> oleh parse engine akan dikonversi menjadi bentuk *caselessLiteral*

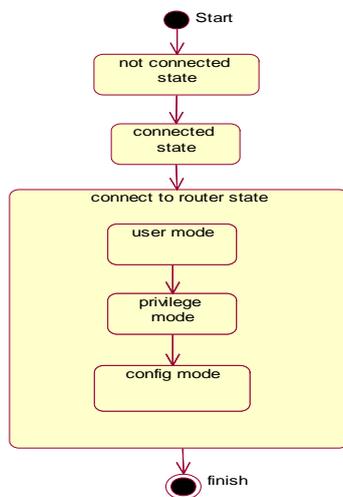
sehingga inputan dalam bentuk huruf besar atau kecil tetap dapat diterima. Kemudian merujuk pada perintah string di atas subparser <target> selalu merupakan "single word", sehingga penentuan rule pada subparser target menjadi seperti ini.

notConnected = crackCommand | scanCommand | upgradeCommand

crackCommand = "crack" "target".

Target = word

Selanjutnya, rules dari desain grammar di atas dapat dilengkapi dengan mengulang langkah sebelumnya terhadap parser yang lain. Penggunaan metode top-down parsing dalam pengkodean sintaks game dapat dibagi menjadi empat state yang masing-masing state memiliki grammar unik dan berbeda. Berikut ini disajikan state player saat memainkan game dalam bentuk statechart diagram



Gambar 7: Statechart Diagram pada Frame Utama

State not connected, state connected dan state connect to router dari gambar 7 di atas masing-masing penjelaskannya dijabarkan dalam tabel 1.

Tabel 1: State dalam Game Melacak Koruptor

No	State/ Keadaan	Keterangan
1	Sebelum terkoneksi	Dalam mode ini komputer player belum terkoneksi dengan komputer target atau router target.
2	Ter-koneksi Dengan komputer	Dalam mode ini player berhasil masuk kedalam komputer target dan dapat melakukan apapun termasuk mendownload dokumen.
3	Ter-koneksi dengan router	Dalam mode ini materi cisco router yang sesungguhnya dilakukan. Pemain akan memanfaatkan berbagai sintaks cisco pada mode ini untuk memenangkan permainan.

### 3.4.1 Keadaan Belum Terkoneksi (Not Connected State)

Dalam mode yang pasti ditemui player pada setiap awal permainan ini, player dapat memasukkan beberapa perintah kedalam panel command console. Perintah-perintah ini akan dieksekusi jika memenuhi standard rule grammar yang sudah didefinisikan sebelumnya. Adapun beberapa perintah yang tersedia dalam mode ini dipaparkan dalam tabel 2 sebagai berikut.

Tabel 2: Perintah grammar pada mode belum terkoneksi

No	Input sintaksis	Fungsi
1	help	Berfungsi untuk penampil bantuan kedalam layar
2	crack [target]	Melakukan proses perusakan keamanan terhadap komputer target
3	Upgrade [device]	Melakukan upgrade device
4	Scan [target]	Melakukan pemindaian detail target, seperti alamat, jumlah bit enkripsi.
5	telnet [target] [password]	Sintaks untuk masuk ke dalam komputer target

### 3.4.2 Keadaan Terkoneksi (Connected State)

Player akan masuk dalam keadaan ini setelah dapat melakukan crack target dan mengetikkan perintah telnet terhadap target. Dalam keadaan terkoneksi ini player dapat melakukan perintah download, upload, transfer, ls dan logout. Berikut ini beberapa perintah dari player kedalam panel command console. Perintah-perintah ini akan dieksekusi jika memenuhi standard rule grammar yang sudah didefinisikan sebelumnya. Adapun beberapa perintah yang tersedia dalam mode ini dipaparkan dalam tabel 3 sebagai berikut.

Tabel 3: Perintah grammar pada mode terkoneksi

No	Input sintaksis	Fungsi
1	Download [dokumen]	Melakukan pemindahan file dari komputer target menuju komputer player
2	Ls	Melihat daftar file yang tersedia dalam komputer target
3	Transfer [jumlah]	Melakukan proses pemindahan uang dari target ke dalam rekening player
4	logout	Memutuskan hubungan koneksi dengan target
5	cat	Membuka dokumen yang berhasil didownload

## 4. Hasil dan Pembahasan

Alur dari sistem aplikasi game Melacak Koruptor dapat dijelaskan dengan tampilan halaman program yang sudah dibuat: Game ini memanfaatkan game engine GTGE untuk menangani proses grafisnya. GTGE memiliki aturan penulisan program tersendiri, agar dapat

berjalan dengan baik, game ini juga mengikuti aturan penulisan program dari GTGE yaitu sebagai berikut [6]:

- a) Class mainInterface extend GameObjek:  
Class ini adalah class utama yang digunakan sebagai host untuk menempatkan class yang lain
- b) Public mainInterface:  
Adalah constructor dari mainInterface yang menjadikan class mainInterface sebagai anak dari class induk cisco-hack
- c) Public method initResources():  
Method ini berfungsi untuk meletakkan background, music dan pemacu awal permainan.
- d) Public method update():  
Method ini merupakan method kedua yang berfungsi untuk mengupdate perubahan yang terjadi didalam game.
- e) Method render():  
Dalam method ini semua gambar yang sudah diinisialisasikan di initResource dan semua perubahan game tiap detik akan ditampilkan

Sebelum memulai, user akan diminta melakukan setting resolusi seperti gambar 8 berikut:



Gambar 8. Setting resolusi game

Kemudian akan muncul halaman intro game. Kemudian menu game untuk mengakses ke interface utama akan muncul. Berikut **gambar 9** interface utama tersebut



Gambar 9. Interface Utama Game

## 5. Kesimpulan dan Saran

### 5.1 Kesimpulan

Dari penelitian dan implikasi uji coba penelitian ini, dapat disimpulkan bahwa penggunaan metode top-down parsing untuk menganalisa sintaks yang diinputkan oleh pemain dapat diimplementasikan dengan baik. Selain itu pemanfaatan parse engine sebagai library untuk menganalisa sintaks dapat berkerja dengan baik ketika digabungkan dengan *grammar top-down parsing*.

### 5.2 Saran

Dalam game edukasi cisco ini, masih terdapat beberapa kekurangan seperti kurang tepatnya pemanfaatan GTGE sebagai game engine. Hal ini karena GTGE tidak mendukung adanya class JFrame yang disediakan oleh java. Sehingga peneliti harus membuat sendiri objek sejenis JTextArea yang mutlak dibutuhkan dalam game ini. Level dari game ditambahi dan tiap level dipecah menjadi beberapa sub level. Materi yang dari router cisco dapat diperluas, sehingga bukan hanya konsep routing, subnetting, dan NAT. Untuk penggunaan animasi, disarankan ditambahi dan tidak hanya berfokus pada panel saja. Untuk penelitian lebih lanjut, konsep dari game ini dapat dikembangkan untuk jenis pembelajaran yang lain dan bukan hanya cisco router.

### Daftar Pustaka

- [1] Bates, Bob. 2004. *Game Design*. Boston: Premier Press
- [2] Rafiudin, Rahmat. 2004. *Mengupas Tuntas CISCO Router*. Jakarta: Elex Media Komputindo
- [3] Salma, Dewi dkk. *Mozaik Teknologi Pendidikan*. 2004. Jakarta: Prenada Media Group
- [4] Utdirartatmo, FIRRAR. 2005. *Teknik Kompilasi*. Yogyakarta: Graha Ilmu
- [5] Utdirartatmo, FIRRAR. 2001. *Teori Bahasa dan Otomata*. Yogyakarta: Graha Ilmu
- [6] <http://goldenstudios.or.id/products/GTGE/>: Diakses pada November 2011 13.14 WIB
- [7] <http://yohan.es/compiler/tutorial1/>: Diakses pada November 2011: 22.34 WIB

### Biodata Penulis

**Ahmad Syamsudin**, lahir di Ponorogo pada tanggal 2 September 1988, memperoleh gelar Sarjana Komputer (S.Kom), Program Studi Teknik Informatika di UIN Maulana Malik Ibrahim Malang, lulus tahun 2011. Tahun 2012 melanjutkan studi di Magister Teknik Informatika STMIK AMIKOM Yogyakarta. Saat ini penulis juga bekerja sebagai Staf Pengajar program Studi Sistem Informasi di Universitas Nusantara PGRI Kediri.